

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2002-202893

(43)Date of publication of application : 19.07.2002

(51)Int.Cl. G06F 9/46
G06F 1/00

(21)Application number : 2001-325363 (71)Applicant : TEXAS INSTRUMENTS INC

(22)Date of filing : 23.10.2001 (72)Inventor : CHAUVEL GERARD
D INVERNO DOMINIQUE
EDWARDS DARVIN R

(30)Priority

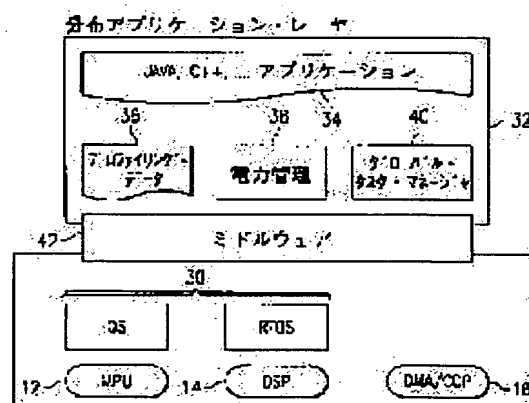
Priority number : 2000 00402948 Priority date : 24.10.2000 Priority country : EP

(54) METHOD FOR CONTROLLING EXECUTION OF MULTIPLEX TASK AND PROCESSING CIRCUIT

(57)Abstract:

PROBLEM TO BE SOLVED: To achieve energy control in a circuit without exerting any significant influence on performance.

SOLUTION: This multi-processor system (10) includes a plurality of processing modules, such as an MPU (12), a DSP (14), and a co-processor/DMA channel (16). Then, a scenario fulfilling such planned power objectives, such as providing maximum operation within package thermal constraints or using minimum energy is built/by using a profile (36) for the various processing modules and a task to be executed and power management software (38). The actual activity associated with the task is monitored during operation, to ensure compatibility with the objectives. It is possible to dynamically change the allocation of tasks by coping with the changes in environment conditions and the changes in task list. The activity information related with various sub-systems is monitored so that it is possible to compute temperatures at various points in the multi-processor system.



LEGAL STATUS

[Date of request for examination] 25.10.2004

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision
of rejection]

[Date of requesting appeal against examiner's
decision of rejection]

[Date of extinction of right]

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号
特開2002-202893
(P2002-202893A)

(43) 公開日 平成14年7月19日 (2002.7.19)

(51) Int.Cl. ⁷	識別記号	F I	テマコード* (参考)
G 0 6 F 9/46	3 6 0	G 0 6 F 9/46	3 6 0 B 5 B 0 9 8
			3 6 0 C
1/00	3 4 0	1/00	3 4 0

審査請求 未請求 請求項の数 3 O L (全 11 頁)

(21) 出願番号 特願2001-325363 (P2001-325363)
(22) 出願日 平成13年10月23日 (2001. 10. 23)
(31) 優先権主張番号 0 0 4 0 2 9 4 8. 4
(32) 優先日 平成12年10月24日 (2000. 10. 24)
(33) 優先権主張国 欧州特許庁 (E P)

(71) 出願人 501229528
テキサス インストルメンツ インコーポ
レイテッド
アメリカ合衆国、テキサス、ダラス、チャ
ーチル ウエイ 7839
(72) 発明者 ジェラルド ショーヴェル
フランス国 アンティーヴ、シエマン デ
ュ ヴァルボスケ 292、レジダンス
デュ ヴァルボスケ 20
(74) 代理人 100066692
弁理士 浅村 皓 (外3名)

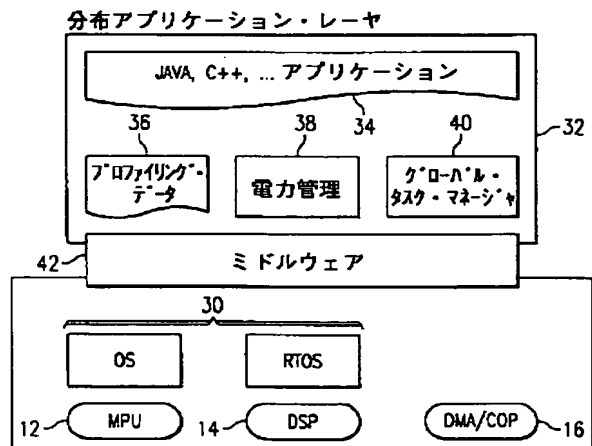
最終頁に続く

(54) 【発明の名称】 多重タスクの実行を制御する方法と処理回路

(57) 【要約】

【課題】 性能に重大な影響を与えずに、回路内のエネルギー管理をする。

【解決手段】 マルチプロセッサ・システム (10) が、MPU (12)、DSP (14) 及びコプロセッサ/DMAチャンネル (16) のような複数の処理モジュールを含む。種々の処理モジュール及び実行すべきタスクに対するプロファイル (36) と共に電力管理ソフトウェア (38) を使って、熱パッケージの制約の範囲内で最高の動作を与えとか最小限のエネルギーを使うとかというような予定の電力目標を満たすシナリオを組立てる。タスクに関連する実際の活動を動作中に監視して、この目標と両立することを確認する。環境状態の変化及びタスク・リストの変更に対処して、タスクの割当てをダイナミックに変えることが出来る。種々のサブシステムに関連する活動情報を監視することにより、マルチプロセッサ・システム内の種々の点に於ける温度を計算することが出来る。



【特許請求の範囲】

【請求項 1】 幾つかのモジュールを含む処理回路に於ける多重タスクの実行を制御する方法に於て、前記処理回路の種々の区域に於ける温度に関連する情報を決定し、

1 つ又は更に多くの区域が温度閾値を超える問題を防止する為に、前記温度に関連する情報に回答して、前記複数個の処理モジュールでタスクを実行する工程を含む方法。

【請求項 2】 多重タスクを実行する複数個の処理モジュールを含む処理回路に於て、前記処理回路の種々の区域に於ける温度に関連する情報を決定する回路と、

1 つ又は更に多くの区域が温度閾値を超えることに伴う問題を防止する為に、前記温度に関連する情報に回答して、前記複数個の処理モジュールでタスクを実行する回路とを含む処理回路。

【請求項 3】 信号を受信及び送信するアンテナと、前記アンテナに結合されていて、オーディオ及びデータ信号を送信及び受信する受信／送信回路とを含み、前記受信／送信回路は、処理回路を含んでおり、前記処理回路は、前記処理回路の種々の区域に於ける温度に関連する情報を決定する回路、及び 1 つ又は更に多くの区域が温度閾値を超えることに伴う問題を防止する為に、前記温度に関連する情報に回答して、前記複数個の処理モジュールでタスクを実行する回路を含む移動通信装置。

【発明の詳細な説明】

【0001】 この出願は 2000 年 10 月 24 日に出願されたヨーロッパ出願通し番号 00402948、4 (T I - 3 1 3 6 2 E U) の優先権を主張する。

【0002】

【発明の属する技術分野】 この発明は全般的に集積回路、更に具体的に言えば、プロセッサのエネルギー管理に関する。

【0003】

【従来の技術及び課題】 長年の間、マイクロプロセッサ・ユニット (MPU)、コプロセッサ及びデジタル信号プロセッサ (DSP) の設計を含むプロセッサの設計の焦点は、プロセッサの速度及び機能性を高めることであった。現在、エネルギー消費が重大な問題になっている。速度及び機能性に重大な障害とならずに、エネルギー消費を小さく保つことが、多くの設計で前面に押し出されていることが重要である。スマート・ホン、携帯電話、PDA (パーソナル・デジタル・アシスタント) 及びハンドヘルド・コンピュータのような多くのシステムは、比較的小さい電池で動作するので、多くの用途ではエネルギー消費が重要になっている。短い期間で電池を再充電するのは不便であるので、こういうシステムでは電池の寿命を最大にすることが望ましい。現在、エネルギー消費を最小限に抑える方式は、スタティック・エネ

ギ管理、即ち使うエネルギーが少ない回路を設計することである。場合によっては、休止期間の間、クロック速度を下げたり、回路を不動作にするようなダイナミックな措置も取られている。こういう変更も重要であるが、特にデバイスを使う便宜の点で、寸法、従って電池の寸法が重要であるようなシステムでは、エネルギー管理を常に改善することが必要である。全体的なエネルギー節約の他に、複雑な処理の場合には、集積回路から熱を散逸する能力が 1 つの因子になっている。集積回路はある量の熱を散逸するように設計されている。タスク (アプリケーション・プロセス) が集積回路の幾つかのシステムを必要として、高いレベルの電流を流す場合、回路が過熱し、システム故障又は誤った挙動の原因になることがある。

【0004】 将来、集積回路によって実行されるアプリケーションは更に複雑になるであろうし、1 つの集積回路内にある MPU、DSP、コプロセッサ及び DMA チャンネルを含む幾つかのプロセッサ (以下「マルチプロセッサ・システム」) によるマルチ処理を必要とする可能性がある。DSP は、幾つかの併行するアプリケーションを支援するように発展しようとし、このアプリケーションのあるものは特定の DSP プラットフォームに専用ではなく、インターネットのようなグローバル・ネットワークからロードされることになる。これは、厳しいコストの制約から、小形でピン数の少ない、低コストのパッケージ技術を使うことを必要とする無線マルチメディア機器の分野では特にそうである。従って、過熱を起こさずにマルチプロセッサ・システムが扱うことが出来るであろうタスクは予測できなくなる。従って、性能に重大な影響を与えずに、回路内のエネルギー管理する方法と装置に対する必要が生じている。

【0005】

【課題を解決する為の手段及び作用】 この発明は、幾つかのモジュールを含む処理回路に於ける多重タスクの実行を制御する方法と装置を提供する。処理回路の種々の区域に於ける温度に関連する情報を決定する。1 つ又は更に多くの区域が温度閾値を超えることに伴う問題を防止する為に、温度に関連する情報に回答して、処理回路の複数個の処理モジュールでタスクが実行される。この発明は従来に比べて、装置の種々の区域の温度又は推定温度に基づいて、完全にダイナミックなエネルギー管理を提供することにより、かなりの利点を提供する。装置内で実行されるタスクが変わるとき、エネルギー管理は新しいシナリオを組立てて、装置のどの区域でも温度閾値を超えることがないように保証することが出来る。この発明並びにその利点が更に完全に理解されるように、次に図面について説明する。

【0006】

【実施例】 この発明は図面の図 1-13 についてみれば一番よく理解される。図面全体に互に、同じ要素には同

じ参照数字を使っている。図1は、MPU 12、1つ又は更に多くのDSP 14及び1つ又は更に多くのDMAチャンネル又はコプロセッサ（これを包括的にDMA／コプロセッサ16として示してある）を含む一般的なマイクロプロセッサ・システム10の全体的なブロック図を示す。この実施例では、MPU 12はコア18及びキャッシュ20を含む。DSP 14は処理コア22及びローカルメモリ24（実際には別々の命令及びデータ命令を使うことが出来るし、或いは一体にしたデータ及びメモリを使ってもよい）を含む。メモリ・インターフェース26が共有メモリ28をMPU 12、DSP 14又はDMA／コプロセッサ16の内の1つ又は更に多くに結合する。各々のプロセッサ（MPU 12、DSP 14）は現実のマルチプロセッサ・システムでは、それ自身のオペレーティング・システム（OS）又は実時間オペレーティング・システム（RTOS）の下に完全に自律的に動作することが出来るし、或いはMPU 12が共有の資源及びメモリの環境を監視するグローバルOSを運用することが出来る。図2はマルチプロセッサ・システム10に対するソフトウェア・レーヤ線図を示す。図1に示すように、MPU 12がOSを実行し、DSP 14がRTOSを実行する。OS及びRTOSはソフトウェアのOSレーヤ30を構成している。分布アプリケーション・レーヤ32がJAV A（登録商標）、C++及びその他のアプリケーション34と、プロファイリング・データ36を使う電力管理タスク38とグローバル・タスク・スケジューラ40とを含む。ミドルウェア・ソフトウェア・レーヤ42が、OSレーヤ30と分布アプリケーション・レーヤ32内のアプリケーションとの間を連絡する。

【0007】図1及び2について、マルチプロセッサ・システム10の動作を説明する。マルチプロセッサ・システム10は種々のタスクを実行することが出来る。マルチプロセッサ・システム10の典型的なアプリケーションはスマートホン・アプリケーションで、この場合、マルチプロセッサ・システム10は、無線通信、ビデオ及びオーディオ圧縮解除及びユーザ・インターフェース（即ちLCDの更新、キーボードの復号）を取扱う。このアプリケーションでは、マルチプロセッサ・システム10に組込まれた異なるシステムが、優先度の異なる幾つかのタスクを実行する。典型的には、OSは、組込まれた種々のシステムに対する異なるタスクのタスク・スケジューラ作りを行う。この発明は、タスクのスケジューラを作るときの判断基準として、エネルギー消費を積算する。好ましい実施例では、分布アプリケーション・レーヤ32の電力管理アプリケーション38及びプロファイル36を使って、タスクのリストを実行する為に、確率論的な値に基づくシステム・シナリオを組立てる。シナリオが予定の判断基準に合わない場合、例えば、電力消費が大きすぎる場合、新しいシナリオを作成する。受

容れることが出来るシナリオが確立された後、OSレーヤがハードウェアの活動を監視して、シナリオで予測した活動が正確であったことを検証する。受容れることの出来るタスク・スケジューラ作りのシナリオに対する判断基準は、装置の性質に応じて変わることがある。移動装置に対する1つの重要な判断基準は、エネルギー消費量が最小であることである。前に述べたように、電子通信装置が更に小形化されると、一層小さい電池を割当てることが、エネルギー消費にプレミアムを付けることになる。多くの場合、装置の動作中、電力を減らす為に、特に電池が低レベルに達したとき、タスクに対する動作モードが縮小することを許すことが出来る。例えば、LCDの更新速度を下げることは、画質は犠牲にするけれども、電力を減らす。別の選択は、性能が遅くなるという犠牲は払うが、電力を減らす為に、マルチプロセッサ・システム10のMIP（毎秒の百万単位の命令）を減らすことである。電力管理ソフトウェア38は、縮小した性能の異なる組合せを使う異なるシナリオを解析して、受容れることが出来る装置の動作に達する。

【0008】電力を管理するときのもう1つの目標は、所定の電力限界の設定に対し、最高のMIP又は最も少ないエネルギーを見付けることであることがある。図3a及び3bは、マルチプロセッサ・システム10が平均電力散逸（powerdissipation）限界を超えるのを防ぐ為に電力管理アプリケーション38を使った一例を示す。図3aでは、DSP 14、DMA 16及びMPU 12が同時に多数のタスクを運用している。時刻t1に、組込まれた3つのシステムの平均電力散逸が、マルチプロセッサ・システム10に課せられた平均限界を超える。図3bは、同じタスクが実行される場合のシナリオを示しているが、受容れることが出来る平均電力散逸プロファイルを維持する為に、DMA及びDSPのタスクが完了する後まで、MPUのタスクを遅らせる。

【0009】図4aは第1の実施例の電力管理タスク38の動作を示すフローチャートである。ブロック50で、電力管理タスクをグローバル・スケジューラ40によって発動し、これはMPU 12又はDSP 14の内の1つによって実行することが出来る。スケジューラが、近づくアプリケーションを評価し、関連する優先及び除外の規則に従って、それをタスクに分割する。タスク・リスト52は、例えば、オーディオ／ビデオ復号、表示装置制御、キーボード制御、記号認識等を含んでよい。工程54で、タスク・リスト52が、タスク・モデル・ファイル56及び縮小許容ファイル58に照らして評価される。タスク・モデル・ファイル56は分布アプリケーション・レーヤ32のプロファイル36の一部分である。タスク・モデル・ファイル56は、タスク・リストにある各々のタスクに異なるモデルを割当てて前もって作成されたファイルである。各々のモデルが、待ち時間の制約、優先度、データの流れ、基準プロセッ

サ速度に於ける初期エネルギー推定値、縮小の影響、及びMIP及び時間の関数としての所定のプロセッサの実行プロファイルのような関連するタスクの特性を定めるデータの集まりであり、これは実験的に、又は計算機支援ソフトウェア設計技術によって導き出すことが出来る。縮小リスト58は、シナリオを作成するときに使うことが出来る縮小の種類を示している。

【0010】タスク・リストが修正される（即ち、新しいタスクが作られるか或いはタスクが削除される）度に、又は実時間事象が起こったときに、工程54のタスク・リスト52及びタスク・モデル56に基づいて、シナリオが組立てられる。このシナリオは、モジュールに対して種々のタスクを割当て、それらのタスクを実行する優先順位を設定する優先度情報を与える。基準速度に於けるシナリオ・エネルギー推定値59をタスクのエネルギー推定値から計算することが出来る。必要な場合又は望ましい場合、タスクを縮小させることが出来る。即ち、一層少ない資源を使うタスクのモードを、完全版のタスクの代りに使うことが出来る。このシナリオから、ブロック60で活動の推定値が発生される。活動の推定値は、タスク活動プロファイル62（これは分布アプリケーション・レーヤ32のプロファイリング・データ36から）及びハードウェア・アーキテクチャ・モデル64（これも分布アプリケーション・レーヤ32のプロファイリング・データ36から）を使って、このシナリオによって生ずるハードウェアの活動に対する確率論的な値を発生する。確率論的な値は、各々のモジュールの待ち／運用時間の割合（実効MHz）、キャッシュ及びメモリに対するアクセス、I/Oトグル速度及びDMAの流れの要請及びデータの分量を含む。熱時定数と釣り合う期間Tを使って、基準プロセッサ速度に於けるエネルギー推定値59と工程60で得られた平均活動（特に、実行プロセッサ速度）から、平均電力散逸を計算することが可能であり、これを熱パッケージ・モデルと比較する。電力の値が、熱パッケージ・モデル72に定められた何れかの閾値を超えた場合、判定ブロック74で、シナリオを拒絶する。この場合、ブロック54で新しいシナリオが組立てられ、工程60、66及び70が繰返される。そうでない場合、シナリオを使って、タスク・リストを実行する。

【0011】シナリオによって定められたタスクの動作中、OS及びRTOSが、ハードウェアに設けられたカウンタ78を使って、ブロック76で夫々のモジュールの活動を追跡する。マルチプロセッサ・システム10のモジュールに於ける実際の活動は、ブロック60で推定した活動とは違っていることがある。ハードウェア・カウンタからのデータを期間T毎に監視して、活動測定値を発生する。活動測定値をブロック66で用いてこの期間に対するエネルギーの値、従って前述のようにブロック66に於ける平均電力の値を計算し、ブロック72の熱

パッケージ・モデルと比較する。測定値が閾値を超える場合、ブロック54で新しいシナリオを組立てる。活動測定値を連続的に監視することにより、シナリオをダイナミックに修正して、予め定めた限界内に留まるようにするか、或いは変化する環境条件に合わせて調節することが出来る。

【0012】チップに対するTの間の合計エネルギー消費量は次のように計算される。

$$E = \int_T \sum_{\text{modules}} [\alpha \cdot C_{pd} \cdot f \cdot V_{dd}^2] \cdot dt \\ \equiv \sum_{\text{modules}} [\sum_T (\alpha)] \cdot C_{pd} \cdot f \cdot V_{dd}^2$$

ここでfは周波数、V_{dd}は電源電圧、αは確率論的な（又は測定された、この図のブロック76に関する説明を参照されたい）活動である。言い換えると、 $\sum_T (\alpha) \cdot C_{pd} \cdot f \cdot V_{dd}^2$ は等価散逸静電容量C_{pd}によって特長付けられる特定のハードウェア・モジュールに対応するエネルギーであり、カウンタの値は、 $\sum_T (\alpha)$ を示し、EはT内に散逸されるマルチプロセッサ・システム10内の全てのモジュールに対する全部のエネルギーの和である。平均システム電力散逸 $W = E / T$ である。好ましい実施例では、測定された並びに確率論的なエネルギー消費量を計算し、平均電力散逸を期間Tに互るエネルギー消費量から導き出す。大抵の場合、エネルギー消費量の情報の方が一層容易に入手出来る。しかし、測定された及び確率論的な電力消費量から電力散逸を計算することも可能である。

【0013】図4bは、第2の実施例の電力管理タスク38の動作を示すフローチャートである。図4bの流れは図4aと同じであるが、工程50でシナリオ組立てアルゴリズムが発動されたとき（新しいタスク、タスクの削除、実時間事象があるとき）、1つの新しいシナリオを選ぶ代りに、性能の制約に見合うn個の異なるシナリオを予め工程54及び59で計算しておいて記憶し、ダイナミック・ループ内での動作の数を減らすと共に、追跡ループで計算された電力がブロック74での現在のシナリオの拒絶になった場合、一層早く適応出来るようにする。図4bでシナリオが拒絶された場合、ブロック65で、予め計算されている別のシナリオが選択される。その他の点で、動作は図4aに示すものと同じである。

【0014】図5-8は、図3の種々のブロックの動作を更に詳しく示している。組立てシステム・ブロック54が図5に示されている。このブロックでは、タスク・リスト52、タスク・モデル56及び考えられるタスク縮小リスト58を使って、シナリオを作成する。タスク・リストは、マルチプロセッサ・システム10でどのタスクを実行すべきであるかに応じて変わる。図5の例では、3つのタスクが示されている。MPEG4が復号し、無線モデムがデータを受信し、キーボード事象を監視する。実例では、タスクは任意の数の源から来るものであってよい。タスク・モデルは、待ち時間及び優先度

の制約、データの流れ、初期のエネルギー推定値及び縮小の影響のような、シナリオを定めるときに考慮すべき条件を示すものである。その他の条件もこのブロックで使うことが出来る。組立てシステム・シナリオ・ブロックの出力がシナリオ80であり、これは種々のタスクをモジュールと関係付け、各々のタスクに優先順位を割振る。例えば、図5に示す例では、MPEG4復号タスクは16の優先度を持ち、無線モデム・タスクは4の優先度を持っている。ブロック54で組立てられるシナリオは、多数の異なる観点に基づくことが出来る。例えば、熱パッケージの制約の範囲内で最大の性能を与えることに基いて、シナリオを組立てることが出来る。この代りに、考えられる最小のエネルギーを使うことに基いてシナリオを組立てることが出来る。最適のシナリオは、装置の動作の間に変化することがある。例えば、電池を一杯に充電したとき、装置は最高の性能レベルで動作することが出来る。電池の電力がプリセット・レベルより少なくなったとき、装置は、動作を維持する為の考えられる最低電力レベルで動作することが出来る。ブロック54からのシナリオ80を図6に示す活動推定ブロック60で使うことが出来る。このブロックは、マルチプロセッサ・システム10に於ける電力使用量に影響を与える種々のパラメータに対する確率計算をする。タスク活動プロファイル62及びハードウェア・アーキテクチャ・モデル64と関連して、確率論的な確率推定値が発生される。タスク活動プロファイルは、データ・アクセスの種類（ロード／記憶）、異なるメモリに対する発生事象、タスクに使われるブランチ及びループのようなコード・プロファイル及びタスク内の命令に対する命令当たりのサイクル数に対する情報を含む。ハードウェア・アーキテクチャ・モデル64は、（プロセッサ運用／待ち時間の割合のような）推定されたハードウェア活動の計算が出来るようにする、システムの待ち時間に対するタスク活動プロファイル62の影響を何らかの形で記述する。このモデルは、タスクが実施されるハードウェアの特性、例えばキャッシュの寸法、種々のバスの幅、I/Oピンの数、キャッシュが書込みか書戻しか、使われるメモリの種類（ダイナミック、スタティック、フラッシュ等）及びモジュールで使われるクロック速度を考慮に入れる。典型的には、モデルは、データにキャッシュが使えるか／使えないか、読取／書込みアクセスの割合、命令当たりのサイクル数等のような異なるパラメータに対するMPU及びDSPの実効周波数の変動を表す曲線群で構成することが出来る。図6に示す実施例では、各々のモジュールの実効周波数、メモリ・アクセスの回数、I/Oトグル速度及びDMAの流れに対する値が計算される。電力に影響するその他の因子も計算することが出来る。

【0015】電力計算ブロック66が図7に示されている。このブロックでは、ブロック60からの確率論的な

活動又はブロック76からの測定された活動を使って、種々のエネルギーの値を計算し、こうして期間Tに互る電力の値を計算する。電力の値が、マルチプロセッサ・システム10のハードウェアの設計に特定のハードウェア電力プロファイルに関連して計算される。ハードウェア・プロファイルは、各々のモジュールに対するCp d、論理設計形式（D型フリップフロップ、ラッチ、ゲート・クロック等）、電源電圧及び出力に対する容量性負荷を含むことがある。電力の計算は、集積モジュールに対し、並びに外部メモリ又はその他の外部装置に対して行うことが出来る。活動測定及び監視ブロック76が図8に示されている。マルチプロセッサ・システム10の全体に互ってカウンタを設けて、キャッシュ・ミス、TLB（変換ルックアサイド・バッファ）ミス、キャッシュが使えないメモリ・アクセス、待ち時間、異なる資源に対する読取／書込み要請、メモリ・オーバヘッド及び温度のような種々のモジュールの活動を測定する。活動測定及び監視ブロック76が、各モジュールの実効周波数に対する値、メモリ・アクセスの回数、I/Oトグル速度及びDMAの流れを出力する。特定の構成では、この他の値も測定することが出来る。このブロックの出力が電力計算ブロック66に送られる。

【0016】図9は、電力／エネルギー管理ソフトウェアを用いるマルチプロセッサ・システム10の一例を示す。この例では、マルチプロセッサ・システム10が、OSを実行するMPU 12、及び夫々のRTOSを実行する2つのDSP 14（個別にDSP 14a及びDSP 2 14bと示す）を含んでいる。各々のモジュールが、マルチプロセッサ・システム10に互って種々の活動カウンタ78の値を監視する監視タスク82を実行する。電力計算タスクがDSP 14aで実行される。種々の監視タスクが、関連する活動カウンタ78からのデータを検索し、その情報をDSP 14aに送って、測定された活動に基づいて電力の値を計算する。電力計算タスク84及び監視タスク82のような電力管理タスクは、他のアプリケーション・タスクと共に実行することが出来る。好ましい実施例では、電力監視タスク38及びプロファイル36が、JAVA実時間の環境でJAVAクラス・パッケージとして構成されている。図10は、図9に示す種類の処理装置100の分解図で、種々の部品の配置が半導体ダイ102の上に示されている。例えば、MPU 12、DSP 1 14a及びDSP 2 14bという部品の境界が図10ではダイ102の上に示されている。ダイ102は、パッケージ110の中に設けられている。ダイ102の上方に、処理装置100の動作中に起こり得る電力散逸プロファイル112の一例が示されている。電力散逸プロファイル112は、夫々の部品の動作に関連するピーク114、116及び118を示している。この図から判るように、電力散逸ピーク114は予定の安全範囲を超えている。

【0017】電力散逸プロファイル112は、図9に示すような部品に関連した種々のカウンタ78によって検出された事象から計算することが出来る。ダイに対する温度フィールドは散逸電力プロファイルから計算することが出来る。ピーク114に示したような臨界的な電力サージが検出されたとき、電力計算タスク84によって、タスクのスケジュールのやり直しを計算することが出来る。この場合、ピーク114を受容可能なレベルに下げる為に、幾つかの解決策を利用することが出来る。第1に、MPU 12で運用するタスクが優先度の高いタスクであった場合、優先度の低いタスクをDSP114a又はDSP214bにスケジュールし直すことが可能であるかもしれない。DSP114a及びDSP214bで表した区域に於ける電力散逸が、MPU12で表した区域に於ける電力散逸に寄与するから、DSP114a又はDSP214bを使うタスクの1つ又は更に多くのスケジュールのやり直しが、ピークを下げることもある。この代りに、MPU12、DSP114a及びDSP214bの周波数を下げることにより、ピーク114で示した電力散逸を下げる

ことが出来る。カウンタ78は、ダイ102の多くの区域に於ける活動を測定することが出来る。例えば、MPU12では、第1のカウンタが命令キャッシュの活動を測定し、第2のカウンタがデータ・キャッシュの活動を測定し、第3のカウンタがMAC（マルチプライヤ・アキュムレータ）の活動を測定することが出来る。カウンタ78は、その活動を測定する回路の区域内に物理的に配置する必要はない。1個のカウンタが、ダイ102の多数の区域に影響する活動を測定することも可能である。活動の影響は、ダイ102の1つ又は更に多くの区域に於ける電力散逸の推定値に直接的に変換することが出来るから、活動の継続的な測定が、装置の性能に影響を与える危険のある電力サージを確認することが出来る。閾値は、危険な状況を識別するように設定することが出来る。

【0018】図11は、ダイの特定の区域に於ける臨界的な温度の影響を避ける為に事象をスケジュールする動作を示すフローチャートである。工程120で、電力管理ソフトウェアが活動情報を受取る。この情報を使って、工程122で、半導体ダイ102の上に於ける電力散逸の分布を計算する。電力散逸の分布が工程124で解析される。半導体ダイ102の何れかの区域で閾値を超えた場合、工程126で、タスクを調節して、その区域に於ける電力散逸を減らす。活動の測定値を用いて電力散逸の分布が推定されるが、半導体上の種々の点に於ける温度を直接的に測定し、実際の温度測定値に基づいてタスクをスケジュールすることが可能である。例えば、PN接合のI-V特性の測定された変化により、温度を推定することが出来る。装置100の動作中、種々の部品に於ける活動を監視する他に、カウンタ78を使

って、タスクを実行する間のダイ102の任意の区域に於ける臨界的な温度を避けるスケジュールを作る為に、区域の特定の温度に対してタスクのプロファイルを決めるのに必要な情報を取出すことが出来る。これは図4a及び4bに示すように実施することが出来、装置100の種々の区域に閾値を適用する。この発明のこの一面は、従来に比べて重要な利点である。第1に、これによって、装置の種々の区域の温度又は推定温度に基づく完全にダイナミックな電力管理が出来る。装置100で実行されるタスクが変化するとき、電力管理が新しいシナリオを組立てて、装置のどの区域でも温度閾値を超えないように保証することが出来る。電力管理ソフトウェアは、それが管理する種々のタスクに対してトランスペアレントである。即ち、特定のタスクが電力管理が出来ないものであっても、電力管理ソフトウェアは、装置100の電力容量に見合う形で、タスクを実行する責任を負う。図11は、特定のタスクの動作に関するエネルギー情報を正確に測定するこの発明の実施例を示す。タスクに関連するエネルギー情報の精度を高めることにより、提案されたグローバル・シナリオに対する成功の確率も同様に高くなる。

【0019】図12は、MPU12の更に詳しいブロック図である。MPUコア130がタスクIDレジスタ132及び比較回路134を含む。コア130が命令キャッシュ20a及びデータ・キャッシュ20bに結合される。カウンタ78がコア内の活動を監視する。カウンタ78は、比較回路134の出力に結合された付能ポート（En）を持っている。独立にタスクを実行するかもしれない各々のプロセッサ（即ち、MPU又はDSPのような「自律的な」プロセッサであるが、一般的にはコプロセッサ又はDMAの物理的なチャンネルはそうではない）は、タスクIDレジスタ132及び比較回路134を持つことが出来る。この為、図9に示す装置は、MPU12、DSP114a及びDSP214bに対応する3つのタスクIDレジスタ132を持つことが出来る。動作について説明すると、マルチプロセッサ・システム10によって実行される各々のタスクは独特の確認タグ、即ち、タスクIDを持っている。タスクが実行されるとき、そのタスクIDがタスクIDレジスタ132に記憶される。システムのエネルギー消費量の正確な測定値が特定のタスクに対して測定されているとき、特定のタスクのタスクIDが比較器134にロードされる（監視されるタスクのタスクIDは比較器134又はレジスタ又は他のメモリに記憶することが出来る）。比較器134は、タスクIDレジスタにある識別子が比較器134にロードされたタスクIDと合うとき、第1の論理信号（例えば論理1）を出力する。同様に、タスクIDレジスタの識別子が比較器134にロードされたタスクIDと異なるとき、第2の論理信号（例えば論理0）を出力する。比較器134の出力が装置10にある種々

のカウンタの付能ポートに結合される。比較器 134 で符合があるとき、カウンタはタスクに関連する活動を測定することが出来る。符合しないとき、カウンタは不動作にされ、他のタスクに関連する活動は測定されないようになっている。幾つかのプロセッサの間で、あるハードウェア・システムが共有になっている。この為、個別のタスクに帰因するこういう共有システムの活動を正確に測定する為に、幾つかのカウンタを夫々の比較回路 134 に結合することが出来る。この代りに、共有のハードウェア・システムのカウンタにタスク ID レジスタ及び比較器を設けて、あるタスクが作用しているときだけ、計数を許すことが出来る。

【0020】この実施例は、タスク・プロファイルに対するエネルギー情報を発生する為に使うことが出来る。エネルギー情報は、装置 10 をエネルギー情報を収集する為に動作させている場合は、「オフライン」モードで収集することが出来るし、装置 10 の実際の動作中にこの情報が収集される場合は、「オンライン」モードで収集することが出来る。タスク・プロファイル 36 は、タスクを作り且つ終了するとき、スケジュールを改善する為の装置の動作中、ダイナミックに更新する。エネルギー・プロファイルを作る他に、上に述べたタスクに特定の事象監視能力を他の目的の為に使うことが出来る。このような 1 つの目的は、性能をよくしたデバッグ技術を提供することであろう。例えば、所定のタスクに対し、活動カウンタがある値に達したとき、ブレークポイントをセットすることが出来る。装置 10 の動作中、各々の自律的なプロセッサに対し、現在のタスクのタスク ID をタスク ID レジスタ 132 に記憶する。多重タスク・システムでは、プロセッサが各々の現在のタスクの間で切り変わり、全ての現在のタスクが同時に実行されているような観を生ずる。各々のタスクがプロセッサによってロードされるとき（「活動する」タスク）、種々の状態情報がプロセッサに復元される。図 12 に示す実施例では、活動するタスクに関連するタスク ID が、このタスクに対する状態情報が復元されるとき、タスク ID レジスタ 132 に記憶される。自律的なプロセッサに対するタスク ID レジスタ 132 の内容が、監視されるタスクのタスク ID と符合する時間の間、カウンタ 78 が付能されて、活動情報を蓄積する。プロセッサが異なるタスクに切り換わるとカウンタは活動を無視する。この為、多重タスク動作中のあるタスクに関連する活動に関する正確な情報を求めることが出来る。

【0021】図 12 に示す実施例は、図 10 及び図 11 に示す実施例と共に使って、希望により、区域に特定の温度データを求めることが出来る。この発明のこの一面は、更に正確なプロファイル・データが得られ、これはタスクをスケジュールする為に使うことが出来る。一層よいエネルギー情報を提供することにより、図 4a 及び 4b に関連して説明したグローバル・シナリオを計算する

成功率が高くなる。

【0022】図 13 は、マイク 152、スピーカ 154、キーパッド 156、表示装置 158、及びアンテナ 140 を持つ移動無線装置 150 の構成を示す。内部処理回路 162 が、ここで述べたエネルギー節約の特長を持つ 1 つ又は更に多くの処理装置を含む。勿論、他の色々な形式の通信装置及びコンピュータ・システムも、特に電池の電力に頼るものは、この発明から利益を生むことが出来ると考えられている。このような他のコンピュータ・システムの例としては、パーソナル・ディジタル・アシスタント (PDA)、ポータブル・コンピュータ、スマートホン、ウェブホン等がある。デスクトップ及び線路から給電されるコンピュータ・システム及びマイクロコントローラの用途でも、特に信頼性の観点から、電力散逸は関心があるから、この発明はこのような線路から給電されるシステムにとっても利点をもたらすと考えられる。電気通信装置 150 が、オーディオ入力を受取るマイク 152、及びオーディオ出力を出力するスピーカ 154 を普通のように具えている。マイク 152 及びスピーカ 154 が、オーディオ及びデータ信号を受信及び送信する処理回路 162 に接続されている。この発明の詳しい説明はある実施例を対象としているが、これらの実施例の種々の変更並びにこの他の実施例も、当業者に容易に考えられよう。この発明は、特許請求の範囲内に属するあらゆる変更又は代りの実施例をも含むものである。

【0023】以上の説明に関し、更に以下の項目を開示する。

(1) 幾つかのモジュールを含む処理回路に於ける多重タスクの実行を制御する方法に於て、前記処理回路の種々の区域に於ける温度に関連する情報を決定し、1 つ又は更に多くの区域が温度閾値を超える問題を防止する為に、前記温度に関連する情報に応答して、前記複数個の処理モジュールでタスクを実行する工程を含む方法。

(2) 第 1 項に記載の方法に於て、前記決定する工程が、前記モジュールによって実行される動作を監視する工程を含む方法。

(3) 第 1 項に記載の方法に於て、前記決定する工程が、前記処理回路内の種々の場所に於ける電力散逸情報を計算する工程を含む方法。

(4) 第 1 項に記載の方法に於て、前記決定する工程が、前記処理回路内の種々の場所に於ける現在の温度を計算する工程を含む方法。

(5) 第 1 項に記載の方法に於て、前記決定する工程が、タスク割当てシナリオを作成し、前記処理回路内の種々の場所に対する温度に関連する情報を推定し、この活動に伴う温度を計算する工程を含む方法。

(6) 第 5 項に記載の方法に於て、タスク割当てシナリオを作成する前記工程が、実行すべきタスクを記述するタスク・リスト及びタスクを記述するタスク・モデル

を受取る工程を含む方法。

(7) 第6項に記載の方法に於て、前記タスク・モデルが各々のタスクに対する初期の区域特定電力散逸推定値を含む方法。

【0024】(8) 多重タスクを実行する複数の処理モジュールを含む処理回路に於て、前記処理回路の種々の区域に於ける温度に関連する情報を決定する回路と、1つ又は更に多くの区域が温度閾値を超えることに伴う問題を防止する為に、前記温度に関連する情報に

10 応答して、前記複数の処理モジュールでタスクを実行する回路とを含む処理回路。

(9) 第8項に記載の処理回路に於て、前記決定する回路が、前記処理モジュールによって実行される動作を監視する回路を含む処理回路。

(10) 第8項に記載の処理回路に於て、前記決定する回路が、前記処理回路内の種々の場所に於ける電力散逸情報を計算する回路を含む処理回路。

(11) 第8項に記載の処理回路に於て、前記決定する回路が、前記処理回路内の種々の場所に於ける現在の温度を計算する回路を含む処理回路。

(12) 第8項に記載の処理回路に於て、前記決定する回路が、タスク割当てシナリオを作成し、前記処理回路内の種々の場所に対する温度に関連する情報を推定し、この活動に伴う温度を計算する回路を含む処理回路。

(13) 第12項に記載の処理回路に於て、前記タスク割当てシナリオを作成する回路が、実行すべきタスクを記述するタスク・リスト及びタスクを記述するタスク・モデルを受取る回路を含む処理回路。

(14) 第13項に記載の処理回路に於て、前記タスク・モデルが各々のタスクに対する初期の区域特定電力散逸推定値を含む処理回路。

(15) 信号を受信及び送信するアンテナと、前記アンテナに結合されていて、オーディオ及びデータ信号を送信及び受信する受信／送信回路とを含み、前記受信／送信回路は、処理回路を含んでおり、前記処理回路は、前記処理回路の種々の区域に於ける温度に関連する情報を決定する回路、及び1つ又は更に多くの区域が温度閾値を超えることに伴う問題を防止する為に、前記温度に関連する情報に

40 応答して、前記複数の処理モジュールでタスクを実行する回路を含む移動通信装置。

【0025】(16) マルチプロセッサ・システム(10)が、MPU(12)、DSP(14)及びコプロセッサ/DMAチャンネル(16)のような複数の処理モジュールを含む。種々の処理モジュール及び実行すべきタスクに対するプロファイル(36)と共に電力管理ソフトウェア(38)を使って、熱パッケージの制約の範囲内で最高の動作を与えるとか最小限のエネルギーを使うとかというような予定の電力目標を満たすシナリ

オを組立てる。タスクに関連する実際の活動を動作中に監視して、この目標と両立することを確認める。環境状態の変化及びタスク・リストの変更に対処して、タスクの割当てをダイナミックに変えることが出来る。種々のサブシステムに関連する活動情報を監視することにより、マルチプロセッサ・システム内の種々の点に於ける温度を計算することが出来る。活動測定値を使って、ダイ上の現在の電力散逸分布を計算することが出来る。必要であれば、シナリオの中のタスクを調節して、電力散逸を減らすことが出来る。更に、更に正確なプロファイル情報を得る為に、特定のタスクに対して活動カウンタを選択的に付能することが出来る。

【図面の簡単な説明】

【図1】マルチプロセッサ・システムのブロック図。

【図2】マルチプロセッサ・システムのソフトウェア・レーヤ線図。

【図3】マルチプロセッサ・システムに対するエネルギー管理の利点を示す一例のグラフ。

20 【図4a】図2のエネルギー管理ソフトウェアの好ましい実施例の動作を示すフローチャート。

【図4b】図2のエネルギー管理ソフトウェアの好ましい実施例の動作を示すフローチャート。

【図5】図4のシステム・シナリオの組立てブロックの図。

【図6】図4の活動推定ブロックの図。

【図7】図4の電力計算ブロックの図。

【図8】図4の活動測定及び監視ブロックの図。

【図9】活動カウンタを持つマルチプロセッサ・システムのブロック図。

30 【図10】図9のブロック図に示した種類の処理装置の分解図並びに動作中の散逸電力(mW/mm^2)のグラフ。

【図11】タスクをスケジュールする為の区域に特定の温度データを使うことを示すフローチャート。

【図12】タスクに関連する活動の正確な測定値を出す実施例のブロック図。

【図13】この発明を用いた移動通信装置の図。

【符号の説明】

12 MPU

14 DSP

16 コプロセッサ/DMAチャンネル

30 OSレーヤ

32 分布アプリケーション・レーヤ

34 アプリケーション

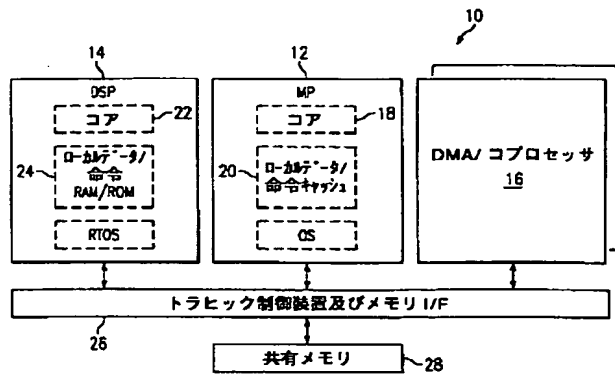
36 プロファイル

38 電力管理

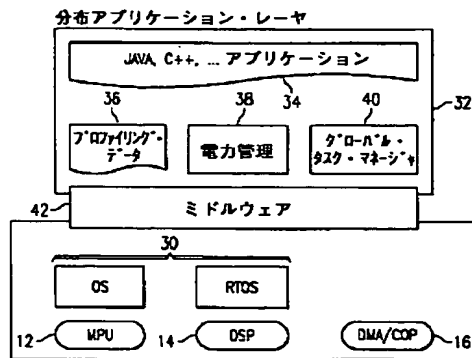
40 グローバル・タスク・マネージャ

42 ミドルウェア

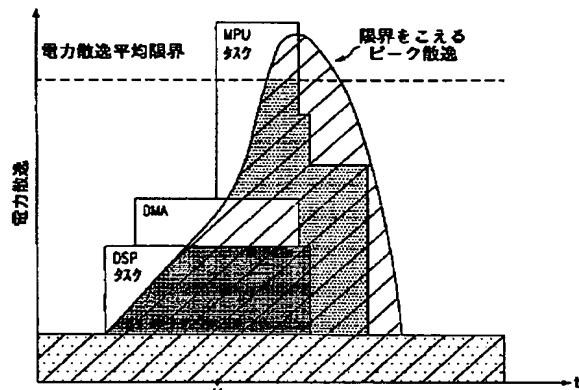
【図 1】



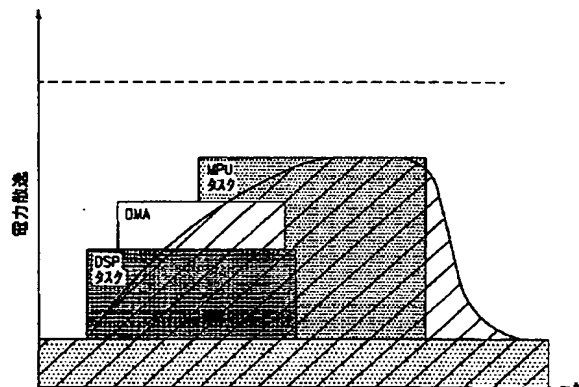
【図 2】



【図 3】

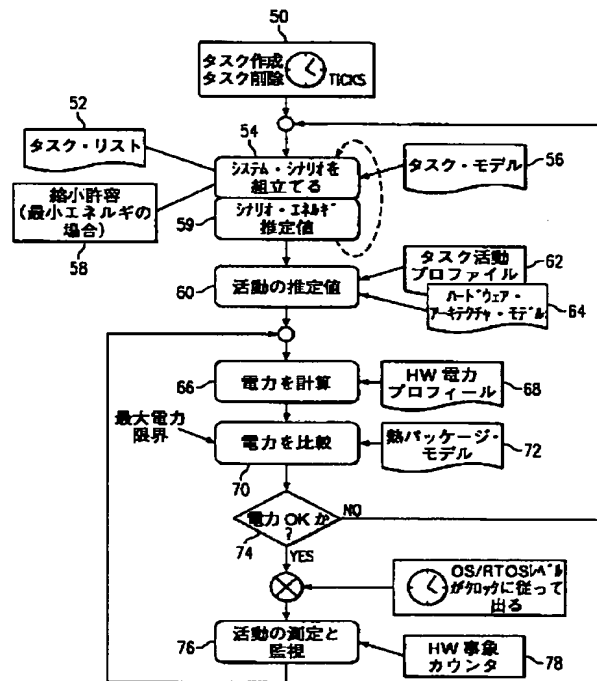


a

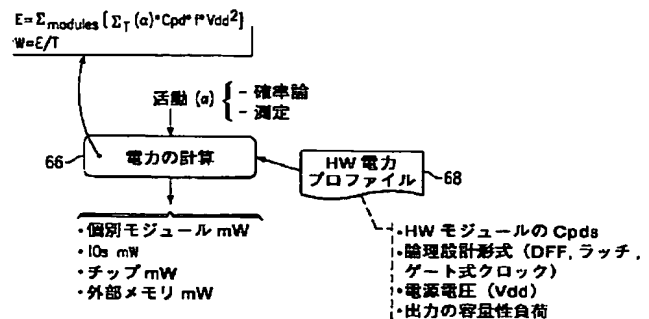


b

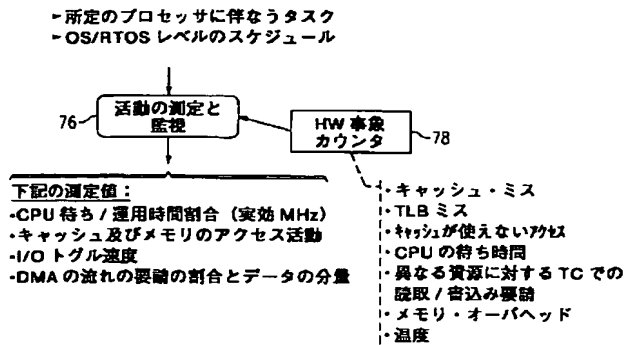
【図 4 a】



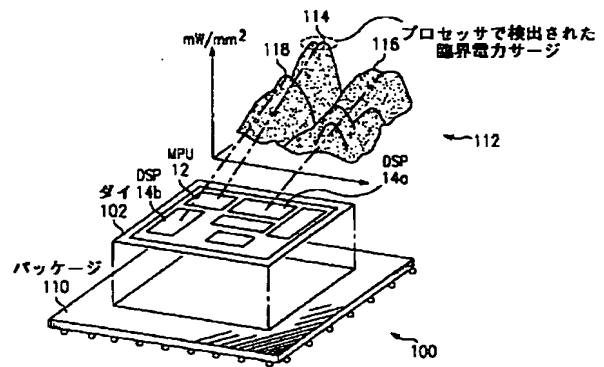
【図 7】



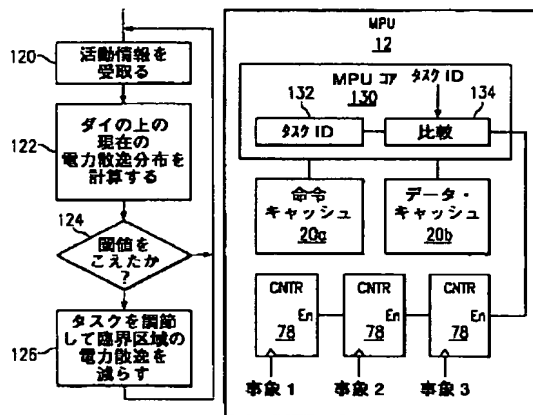
【図8】



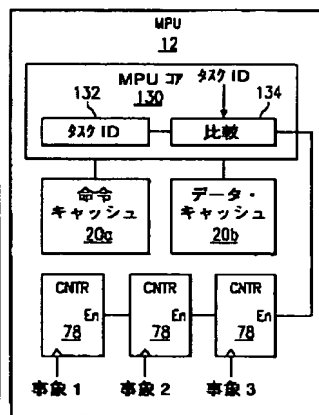
【図10】



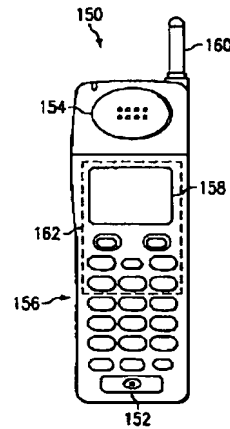
【図11】



【図12】



【図13】



フロントページの続き

(72)発明者 ドミニク ディンヴェルノ
 フランス国 ヴィルヌーヴ - ルーベ、
 シェマン デ バセ ジネスティエール、47

(72)発明者 ダーヴィン アール、エドワーズ
 アメリカ合衆国 テキサス、ガーランド、
 グレン ヴィスタ ドライヴ 5301
 F ターム(参考) 5B098 AA09 AA10 GA02 GA04 GC00
 GC10 JJ08